ORACLE DATABASE SECURITY

Cristina-Maria Titrade[1]

Abstract

This paper presents some security issues, namely security database system level, data level security, user-level security, user management, resource management and password management.

Security is a constant concern in the design and database development. Usually, there are no concerns about the existence of security, but rather how large it should be. A typically DBMS has several levels of security, in addition to those offered by the operating system or network. Typically, a DBMS has user accounts that require a login password to be authenticated to access the data.

Keywords: data security, password administration, Oracle HTTP Server, OracleAS, access control.

Introduction

Oracle database contains its own security system that prevents unauthorized access to the database. The Oracle database security is achieved by users of the database. Database server requires a username and a password for each access to the database, regardless of tool used to interface, the database server does not allow access to the database if it is not used a name and a correct password.

The access of a user to database is given by a number of rights called privileges of the system or database level privileges, which allow the user to perform operations such as connecting to the database and creating objects. Once a user created database objects, it is then responsible for granting rights for other users to objects that are owned by. These rights are called object-level privileges.

Data security

In Oracle, data security is ensured primarily through user data (in the sense used by Oracle, a database user is actually an account and do not identify with a person who accesses the database, in general, many people may use the same account access). Any database object is owned by a user, a user scheme including all objects that it owns. To access an object, a user must either be the owner of that object or have the necessary privileges for this operation. Privileges can be granted directly to a user or can be grouped into roles, which in turn may be granted to the user.
A scheme is a collection of objects available to the user. Scheme objects are logical structures that actually refers to data from a database such as tables, views, sequences, indexes, synonyms, etc.. Each user of the database is granted with certain rights known as

[1] Assistant teacher at the Romanian-American University in Bucharest and Ph.D. student at the Academy of Economic Studies in Bucharest. E-mail: cristina_titrade@yahoo.com

privileges. A privilege is permission to perform an action or access an object which belongs to another user. In Oracle, a user can not execute any action without having the privilege to do so.

Roles are used to simplify the administration of privileges. Thus, instead of a privilege granted directly to user, privileges are granted to a role and a role is granted to a user. In other words, roles are a group of privileges.

Data security is an important function of a database system that protects data against unauthorized access. Data security has two components: data protection and authorized access control.

System level security

System level security requires proper management of users, setting the authentication methods for users and setting the host operating system security. Each database has one or more administrators who are responsible for providing security policies. If the database is small, database administrator may also have security responsibilities.

The system provides several methods of authentication for database users, using passwords, the host operating system, network services or the SSL protocol (Secure Sockets Layer), proxy authentication and authorization.

At the operating system level which resides the server and database applications, the administrator must have the privileges to create and delete files, regular users of database must not have the privilege to create or delete the relevant files of the database operation, security administrators must have the privileges to change security domain for the available accounts in the operating system (if roller identification for Oracle database users is done by operating system).

Data-Level Security

Data level security include mechanisms to control access to data and the use of database objects. There are establish the users who have access to an object and the types of actions allowed on it. In this sense, database users are granted with some system and object privileges, which can be grouped into rolls. Also, for each object of a scheme are defined the actions to be audited.

Another way of establishing security at this level is to use the views. They can restrict access to data of a table, by excluding some columns or rows. The system allows the implementation of security policies by using functions and linking them to tables or views (fine-grained access control). Such a function automatically generates a WHERE condition in a SQL statement, and thus restrict access to certain data lines.

User Level Security

There are two general ways of establishing user-level security: through passwords and granting privileges. If user authentication is managed by the database, then administrators

should develop security policies for passwords. For example, to impose user database to change passwords from time to time, the length of passwords to be large enough and in their structure to enter both letters and numbers.

For databases with many users, administration of privileges is more effective if you use roles. Instead, when the number of users is low, it is advisable to give explicit permission for each user and avoid the use of roles. The administrator must decide which are the categories of user groups and assign roles for each group. Also must decide what privileges must be nominal granted to users.

Security administrators must develop security policies for database administrators too. For very large databases, which require more administrators, security administrator must decide which are the groups of administration privileges and to include them in the administration's role. For small databases is enough to have a single specific role and give it to all database administrators.

Authorized access control in centralized systems

In authorized access control are involved three main actors: users, which triggers the execution of applications, operations that are included in applications, database objects on which operations are performed.

Authorized access control consists in the validation of a triplet: (user, operation, object). The authorization may be seen as a triplet (user, type of operation, the definition of the object) that specifies if the user has the right to perform an operation of that type on the object. To control the correct authorization, a DBMS requires the definition of users, objects and rights.

Entering a user (a person or group of persons) in the system is usually done by a pair (username, password). User name uniquely identifies the user, while the password, known only by the user, authenticate the user. Both are required to connect to the system.
Objects to be protected are subsets of the database. In a relational system objects can be defined by type (views, relations, tuples, attributes) and their contents by using selection predicates. Also, the views mechanism allows protecting objects by simply hiding some subsets of relations (attributes or tuples).

Managing Users and Resources

Depending on the licenses received for Oracle system, must be limited the number of concurrent sessions and the users connect to the database. This is accomplished by setting the initialization parameters of LICENSE type.

Licenses for the use of competition limits the number of sessions that can be simultaneously connected to the database. The maximum number of concurrent sessions (LICENSE _MAX_SESSIONS) can be specified before starting the instance and changed while the database is started. When this limit is reached, the system will send the user a

message announcing it. In this case, only those users with RESTRICTED SESSION privilege can connect to the database.

Limiting the number of users restrict the number of individual approvals for use of the system. Specify the maximum number of users that can be created in the database is done before starting an instance (LICENSE_MAX_USERS). This limit can be changed during operation of the instance, using the ALTER SYSTEM command. After overcoming them, new users can not be created. In this case, the system sends a message that announces that the maximum number of users allowed was reached.

The V$LICENSE view from data dictionary allows to identify the current settings on the limitations, the current number of user sessions and the maximum number of concurrent sessions reached from the begining of the instance.

User authentication methods

To prevent unauthorized use of a user account (username), the Oracle system performs validation of users in various ways, before they initiate a session with the database:

- authentication through database, using passwords;
- external authentication through operating system or network services;
- global authentication by SSL security protocol;
- proxy authentication if users connect to the database through an application server.

In general, the same method is used to authenticate all users of the database. However, Oracle system allows addressing all authentication methods within the same database instance.

If the case of authentication through database, management accounts, passwords and validating users are made entirely by the system. For each user are defined passwords for access. For security reasons they can be stored in encrypted format.

If it is used external authentication, then the user accounts are maintained by the system and password administration and user authentication is done through an external service (operating system or a network service, such as Oracle Net). User accounts will be composed of a prefix followed by the name of their accounts from the operating system.
The prefix is set by the initialization parameter OS_AUTHENT_PREFIX. If its value is changed, the accounts using the old prefix are invalid. The evaluation of default parameter is OPS$. A user trying to connect to Oracle database will be authenticated by the operating system. If in this system the user account is name, then connecting to the database is allowed only if the Oracle system contains the user's correspondent account in the database (OPS$name).

The benefits of authentication by the operating system are:

- users can connect to Oracle system in a conventional way, without using a username and password (for example, a user can invoke SQL*Plus utility, allowing direct command SQLPLUS/);

- users authorization control is centralized in the operating system (in the database should not be stored and managed user passwords).

External authentication service over the network is using infrastructures with public keys (PKI). The fundamental elements of these infrastructures are digital certificates, certification authorities and facilities of managing certificates. In the system operation with public keys is required a generation system, movement and an authentification of the keys used by the user. Certification authorities distribute authenticated key certificates. The certificate is a tamper-proof combination of a public key and a particular attribute or the owner. He carries a digital signature of an certification authority in which, thus, confirm the identity of the subject.

Authentication systems that use infrastructures with public keys issue digital certificates to users to authenticate directly to the server.

Public key infrastructure used by Oracle has the following components:

- SSL protocol (independent of platform and application, which provides authentication services, data compression, encryption and data integrity for a range of application-level Internet protocols) for authentication and secure key management;
- Oracle Call Interface utility and PL / SQL functions to use digital signature and its verification using the associate certificate;
- user certificates issued by certification authority that provides a high level of confidence;
- virtual wallets, containing the user's private key, the certificate of authentication and the list of certificates through which the user obtain a high level of confidence;
- Oracle Wallet Manager, a Java application used for managing and editing a virtual wallet of credentials (user protect keys, manage C.509v3 on the Oracle servers, generates pairs of public and private keys, creates applications for certification by the certifying authority, install certificates, certificates of trust configured, open a wallet to access an Oracle PKI, create wallets that can be opened using Oracle Enterprise Login Assistant);
- Oracle Enterprise Security Manager tool for centralized management of privileges;
- Oracle Internet Directory service, which allows users centralized configuration and administration, including security attributes and privileges for their authentication with X.509 certificates. Oracle Enterprise Login Assistant utility to open or close a user's virtual wallet and enabling or disabling an application of secure communication via SSL.

Through the global authentication, users are identified in the database as global users, using the SSL protocol. Managing users is done outside the database, by centralizing them in a directory called directory service. The global roles are defined in the database, but they are granted by the directory service. The advantage lies in the possibility of establishing centralized management of users and roles links to the directory service for

each database that needs to have access. The solution is to create independent visitors scheme, which allows them to use a scheme in common.

In the case of multitier configuration, proxy authentication involves verifying the right of server access to database applications, protecting the identity and privileges of the clients over all levels and checking only the actions carried out in their favor.

The Oracle system offers two forms of proxy authentication:

- if the client is a global application or a global user, then it will be logged by the application server;
- if the customer is a user database, then it will not be authenticated by the server application (clients identity and passwords go through the server applications to the server database where authentication takes place).

Managing users

Every Oracle database contains a list of users. To access the database, a user must perform a database application and connect to an instance of it using a valid account defined in the database.

Creating a user is done through the CREATE USER command. To be entitled to use that command is necessary to CREATE USER system privilege. In general, the security administrator is the only one who has this privilege. Creating a user is to define its identity (name and password), specify the profile, the default table space, using share table space and temporary table space in which the temporary segments are created.
Simplified form of the CREATE USER command is:

```
CREATE USER user_name
IDENTIFIED
{ BY password l EXTERNALLY
l GLOBALLY AS'CN=user_name, other_identification_attributes'}
[DEFAULT TABLESPACE table_space_name]
[TEMPORARY TABLESPACE table_space_name]
[QUOTA {int [{K l M}] l UNLIMITED} ON table_space_name]
[PROFILE profile_name]
[PASSWORD EXPIRE]
[ACCOUNT { LOCK l UNLOCK}]};
```

User name must be unique. A user and a role can not have the same name.

Authentication mode can be achieved through the database (IDENTIFIED BY password), external (IDENTIFIED EXTERNALLY) or SSL (GLOBALLY IDENTIFIED AS 'identification'). The string identification provides an identification to the service director.

Each user must be assigned with a default table space (DEFAULT TABLESPACE) in which the system stores user-created objects, if is not specified a different table space for them. The default table space is SYSTEM.

For each user can be specified a temporary table space. This table space is used for storing temporary segments that are necessary to SQL commands initiated by the user. If is not specified a temporary table space, the system will allocate by default a temporary table space for user which was defined at the creation of the database. If this space was not specified too, the default temporary table space is SYSTEM.

In order to prevent excessive use of database space we can specify some using limits for the table spaces to which the user has access. These limits are specified in QUOTA clause. If table space limit is 0, then the user can not create new objects, and for objects that allready exist in that table space can not allocate more space. UNLIMITED option involves unlimited use for that table space.

Also, when we create a user we can specify a profile for that user. Profiles facilitate the passwords management and the limits of resources use. If no profile is specified, the default one is associated. PASSWORD EXPIRE clause implies that the user must change password at first login to the database. To lock or unlock a user account is used ACCOUNT clause, with options LOCK, respectively UNLOCK.

Managing passwords and resources using profiles

A profile is a set of resource limitations that can be assigned to a database user. Each Oracle database allows definition of a limitless number of profiles. They must be created and administered only if security policy requires that the use of database resources is limited. To use profiles, first we have to create types of similar user groups.

Profiles can be assigned to each user (using the CREATE USER command or ALTER USER) or we can define default profiles that are associated with all users who do not have a specific profile.

To create a profile is required the system privilege CREATE PROFILE. When we create a profile we can explicit the use limits of private resources or password parameters.

The Oracle can authenticate users using information stored in the database. The most important is the authentication information associated with a user password. This is encryptioned and stored in the data dictionary. The user can always change their own password.

To ensure confidentiality of passwords, encryption system allows their connections during network (client / server or server / server).

If this feature is enabled on both the client machine and the server system to encrypt passwords before sending them into the network, using a modified versiunde encryption algorithm DES (Data Encryption Standard).

If the user enters the wrong password for specified number of times the system locks the account associated with it. Depending on how the account was configured may be

automatically unlocked after a specified period of time, or manually, by the database administrator.

After the profile was created, it can be associated to database users. It is not possible for a user to have multiple profiles simultaneously. If a profile is assigned to a user who already has one, the new profile will replace the old one. Combination of profiles does not affect the current session. Profiles can be attributed only to users, and not to a roller or other profile.

Information about users and profiles

The system maintains a series of views in the data dictionary containing information about database users and profiles:

- DBA_TS_QUOTAS (describes the table space quotas for users); USER_PASSWORD_LIMITS (describes the parameters relating to passwords, set by CREATE PROFILE);
- DBA_PROFILES (lists all the profiles together with their limits); USER_RESOURCE_LIMITS (display resource limitations of the current user); RESOURCE_COST (displays the cost of each resource);
- V$SESSTAT (lists statistics about user sessions);
- V$STATNAM (displays the name of the statistics listed by previous view); PROXY_USERS (describes the database users who can assume the identity of other users), etc..

Managing privileges and roles

A privilege is the right to execute certain SQL commands. Privileges include the right to connect to the database, create tables, select lines of another user table, execution of stored procedures created by another user, etc.. Privileges should be granted only if users are absolutely necessary in such activity. Excessive granting of privileges can compromise the security of its database. Privileges can be system type or object type.
System privileges can be classified as:

- system-specific privileges (eg CREATE SESSION, DROP TABLESPACE, ALTER TABLESPACE)
- privileges for proper management of objects in any scheme (eg, CREATE ANY TABLE, DROP ANY INDEX).

A role is a group of related privileges that can be granted or revoked simultaneously to users or other roles.

Using roles allows:

- to simplify administration privileges (rather than grant more privileges to one group of closely related users, we can create a role that contains all the necessary privileges and then grant the role each group member);
- dynamic administration of privileges (if a users group privileges must be changed, we will change their role that contains it and that will automatically be propagated to each user who is assigned to that role);

- selective activation of the privileges (Roles can be selectively enabled or disabled so that it allows a high control of the privileges granted to users).

Roles have the following features:

- users can be granted or revoked using the same commands as with system privileges;
- may include both system privileges and object privileges;
- can be protected using passwords;
- must have a unique name, different from user accounts and other roles in the database;
- there are not contained in any user scheme;

their characteristics can be found in the data dictionary.

Oracle's Middleware Security

Multitier architectures have replaced the client-server applications in terms of preferred channel for access to applications and data processing. There are many reasons for this, including greater scalability, lower costs and more opportunities. Risks that arise when deploying applications on the Internet should not be overlooked. Such risks include limited knowledge of user identity, minimum control systems user behavior and the increased exposure of data to users malevolent attacks that exploit specific features open Internet, such as "worms", script of "cross-site", etc.

Web application developers have been forced to find solutions to such risks. About these security solutions in the middleware applications will be discussed in the following lines. Recent trends that multiply the possible risks that arise in web applications includes: deployment of several software applications through a single portal for business information, increasing the share of Java technology for web application development and the complexity and the need for scalability to run applications.

Through Oracle Application Server 10g, the Oracle provides a security framework for both internal components of OracleAS as well as third part applications running on OracleAS. OracleAS introduces the term Identity Management which provides support for the process of defining and managing user identity applications.

Security services provided by Oracle HTTP Server

Oracle HTTP Server extends Apache with a variety of standard optimizations or specific Oracle (the "modules" added the Apache server). These enhancements include the ability to allow / restrict access to files and services based on user identity authentication standard established by operations through X.509 certificates and by IP address or hostname.

Another important feature of Oracle HTTP Server is the protection of data exchanged between client and server. This is provided by the SSL protocol, which ensures both data integrity and authentication of users and HTTP servers.

Although the Oracle HTTP Server is based on open source Apache Web server, it contains several enhancements that increase security access control. For example, Apache Server restrict access to directories with files with extension. htaccess. Processing these files is disabled by default in Oracle HTTP Server as file processing .htaccess requires both security issues and the decrease of performance. Oracle HTTP Server implements this type of access control through the modules / plug-ins that offer increased security and better performance for authorizing users to access resources.

Java Security in OracleAS

Java and Java 2 Enterprise Edition in particular has become the preferred development environment for many web applications. J2EE defines a security model type Java2 Security Model and a security framework known as the Java Authentication and Authorization Service (JAAS). OracleAS implements this framework through a J2EE compliant JAAS provider. JAAS Provider ensure accessibility to authentication services, authorization and delegation for developers of applications and allows integration of applications in J2EE environments.

OracleAS implementation for JAAS supports both the authorization information stored in OID and a simple implementation of the authorization API using XML as the encoding mechanism. This API allows Java applications to obtain information about users and roles through a secure mechanism from the operating system files.

OracleAS Portal - security features

Oracle 9iAS Portal is a key component of Oracle's product offerings in the category "enterprise portal". Web products in this class allow access to the newly formed business related information from internal networks of organizations. Although it was originally intended market for corporate portals, OracleAS can be configured to allow access to much larger communities, such as the Internet.

This portal allows users to manage applications and content published on the web and to structure the information logically. It also contains numerous tools to create users and keep track of those already existing, as well as their access to OracleAS Portal.
OracleAS Portal provides a secure platform to integrate different applications into one portal, as well as effective management tools in this environment.

OracleAS Portal provides a consistent model for authorization based on individual and group privileges to give users access to applications and content of the portal. It also provides a flexible model to integrate applications into the portal login interface, allowing them to be classified as portlets, applications partner or external applications. OracleAS Portal also supports the type of security audit events via the event registration service.

Conclusions

Security is a critical issue in the case of multitier applications. Oracle Middleware provides a solid framework for this type of web server applications using Oracle HTTP

Server based on Apache, Oracle's J2EE framework and OracleAS Portal. Secure Application Server OracleAS starts from basic services, well-tested and easily configurable provided by Apache, add enhancements such single sign-on, authorization based on the OID and user management, security services and reaches Java2 security and Portal mechanisms to integrate applications. In addition, OracleAS supports a secure access to Oracle database using Oracle Advanced Security. These features ensure that the OracleAS Infrastructure is a smart choice for development and deployment of multitier applications in a secure environment.

References:

1. Ron Ben Natan – "Implementing Database Security and Auditing: Includes Examples for Oracle, SQL Server, DB2 UDB, Sybase", Digital Press, May 02, 2005;
2. Pete Finnigan – "Oracle Security Step-by-Step", SANS Press, April, 2004;
3. Arup Nanda, Donald Burleson – "Oracle Privacy Security Auditing: Includes Federal Law Compliance with HIPAA, Sarbanes Oxley & The Gramm Leach Bliley Act GLB (Oracle In-Focus series)", Rampant Techpress, December 01, 2003;
4. John Abel - "Oracle E-Business Security Suite (Osborne ORACLE Press Series)", McGraw-Hill Osborne mass-media, 1 edition, August 08, 2008;
5. http://www.oracle.com/ro/technologies/security/index.html.